NCRISET-2017 e-ISSN: 2456-3463
International Journal of Innovations in Engineering and Science, Vol. 2, No.6, 2017
www.ijies.net

Design Of Area Efficient Multiplier By Using Modified Booth Algorithm

Juili Borkar
Mtech (VLSI), Student, Department of ETC,
G.H.Raisoni Institute of Engineering & Technology,
Nagpur, India
Email: - juiliborkar@gmail.com

Dr. U.M. Gokhale
Professor, Department of ETC,
G.H.Raisoni Institute of Engineering & Technology,
Nagpur, India
Email: - umgokhale@gmail.com

Abstract— **Multiplier is one of the important elements in most of the digital processing system such as FIR filters, digital signal processors and microprocessors etc. The two important parameters of a multiplier design are its area and speed that are inversely proportional. The speed of a system depends on how a faster an arithmetic operations are performed. The main problem in designing of VLSI circuits are high power consumption, large area utilization and delay which affect the speed of the computation and also results in power dissipation. In general, speed and power are the two essential factors in VLSI design. For solving the issues, a new architecture has been design. In proposed design, two multipliers are used modified booth multiplier and Wallace tree multiplier with Ripple carry adder. Modified booth multiplier is used to reduce number of partial products whereas Wallace tree multiplier is used for fast addition and finally, ripple carry adder is used for final accumulation. This paper presents study of different booth algorithm and design of multiplier by using modified booth algorithm (MBA). Multiplier circuit will be design using Verilog and simulated using Xilinx ISE Simulator.**

Keyword: - Modified booth algorithm, Wallace tree, and ripple carry adder

I. INTRODUCTION

Multiplication is one of the fundamental operations in most signal processing algorithms. The basic multiplication principle is consists of first evaluation of partial products and then accumulation of shifted partial products. As compared with many other arithmetic operations multiplication is more time and power consuming operation. Hence, enhancing the performance and reducing the power dissipation are the two most important design challenges for all application in which multiplier unit dominate system performance and power dissipation. One of the best effective ways to increase the speed of a multiplier is to reduce the number of the partial products. Fast multipliers are used in many digital signal processing (DSP) and multimedia application in which the output data has directed bearing to the accumulation of series of products over a single multiplication operation. Multipliers have large area, long latency and consume considerable power. Thus low power multiplier design has been an important part in low power VLSI system design.

II. LITERATURE SURVEY

Ravindra P Rajput, M. N *S*hanmukha Swamy [1] proposed a design and implementation of high speed 8x8 modified booth encoder multiplier for signed and unsigned numbers .Here carry look-ahead adder and carry save adder are used for increase the multiplier operation and also different simulation output result of 8x8 of modified booth encoder multiplier for signed-unsigned numbers are given in binary form.

Nishat Bano [2] proposed the design and implementation of booth multiplier by using VHDL. It also compares the power consumption and delay of radix-2 and radix-4 booth multipliers. When implemented on FPGA, it is found that the booth multiplier (Radix-4) consumes 22.9% less power than conventional radix-2 multiplier.

K.Nagarjun, S.Srinivas [3] proposed a new concept for multiplication by using modified booth algorithm and reversible logic functions and modified booth algorithm with reversible gate logic are synthesized and simulated by using Xilinx ISE simulator.

Kulvir Singh , Dilip Kumar [4] design high speed and low area Modified Booth multiplier (MBM) by using carry select adder in three stage pipelining technique .The MBM circuit is designed using VHDL language and the circuits are simulated using Xilinx ISE simulator which gives simulation result of multiplication of unsigned and signed number. Hence, both the delay time and area of high speed Modified booth multiplier found to be 51.92ns and 394 slices are reduced to 22.38ns and 377slices respectively using MBM

with carry select adder which proves that proposed MBM design is more efficient than conventional one in terms of area and delay .

CONVENTIONAL MULTIPLIER

Any multiplier design divided into three stages:-

1. Partial products generation stage
2. Partial products addition stage
3. Final addition stage

In multiplier, the speed of multiplication is increase by reducing number of partial product generated in first stage. Many high performance algorithms and architectures have been proposed to accelerate multiplication operation. Figure.1 shows 4x4 multiplication algorithm for a 4 bit multiplicand and 4 bit multiplier where A, B are two inputs and Y is an output.

A = A3 A2 A1 A0 Multiplicand

B = B3 B2 B1 B0 Multiplier

Y= Y7 Y6 Y5 Y4 Y3 Y2 Y1 Y0
(Multiplication of A and B)

				A3	A2	A1	A0	Inputs
			x	B3	B2	B1	B0	
			C	B0 x A3	B0 x A2	B0 x A1	B0 x A0	
		+	B1 x A3	B1 x A2	B1 x A1	B1 x A0		
		C	sum	sum	sum	sum		
	+	B2 x A3	B2 x A2	B2 x A1	B2 x A0			Internal Signals
	C	sum	sum	sum	sum			
+	B3 x A3	B3 x A2	B3 x A1	B3 x A0				
C	sum	sum	sum	sum				
Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0	Outputs

Fig .1:4x4 multiplications

III. PROPOSED METHODOLOGY

In proposed work, we are designing a 16x16 hybrid multiplier which is consists of modified booth algorithm and Wallace tree structure. The figure 2 shows the block diagram of Hybrid multiplier it consists of three stages first stage is modified booth stage, second stage is Wallace tree stage, and third stage is final accumulation stage. Multiplier and multiplicand are considered as two inputs to modified booth stage. If negative number taken as input then it is, represented in 2's complement and if positive number is consider as input then it will keep as it is. Modified booth stage consists of partial product generation by using modified booth algorithm. In modified booth algorithm the reduction of partial product takes places by using efficient encoding method which save multiplier area and reduce delay at the same time .Wallace tree structure is used for fast addition and for final accumulation either ripple carry adder or carry look ahead adder is used.

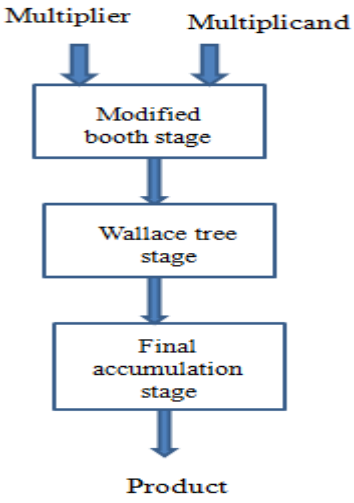

Fig.2: Block diagram of Hybrid Multiplier

IV. MODIFIED BOOTH STAGE

As discuss above the modified booth stage is responsible for generating partial products by using modified booth algorithm. For understanding modified booth algorithm first, we have to understand booth algorithm.

Booth multiplier (Radix-2 algorithm):-

In year 1951, Andrew D. Booth [5] devised a multiplication algorithm, which was named as Booth's algorithm. The booth algorithm multiplies two signed binary numbers in two's complement notation. Booth Multiplier is designed by using booth algorithm. Booth's algorithm is based on recoding the multiplier (Y) to a recoded value (Z) and leaving the multiplicand (X) unchanged. In Booth recoding process, each digit of the multiplier can assume positive as well as negative and zero values. There is a special notation, called signed digit (SD) encoding and used to express these signed digits. In Signed digit encoding +1 and 0 are expressed as 1 and 0, but -1 is expressed as 1[6]. The value of a 2's complement integer was defined a by equation (1).

$$Y = -Y_{m-1}\,2^{m-1} + \sum_{i=0}^{m-2} Y_i\,2^i \qquad -(1)$$

This equation says that in order to get the value of a signed 2's complement number, multiply the m - ith digit by -2 , -1and multiply each remaining digit i by +2g [6].For implementing booth algorithm one of the most important step is booth recoding. By booth recoding, we can replace string of 1's by 0's. Hence, if this number were to be used as the Multiplier bits in a multiplication, we could replace 5 additions by one addition and one subtraction.
The Booth recoding [10] procedure is as follows:-

1. Working from 1st bit to MSB, replace each 0 digit bit of the original number with a 0 in the recoded number until a 1 is encountered.
2. When 1 is encountered, insert a 1 bit at that position in the recoded number, and skip over any succeeding 1's bit until a 0 is encountered.
3. Replace that 0 with a 1 and continue. This algorithm is expressed in tabular, considering pairs of numbers, Yi and Yi-1, and the recoded digit Zi as shown in Table 1.

Table 1: Booth recoding table for Radix -2 [6]

Y_i	Y_{i-1}	Z_{i-i}	Multiplier value	Situation
0	0	0	0	String of 0's
0	1	1	+1	End of string of 1's
1	0	1	-1	Begin string of 1's
1	1	0	0	String of 1's

Booth algorithm (Radix-2):-

1. Add bit 0 to right of LSB of multiplier and look at rightmost bit of multiplier to make pairing of 2 bits from right to left and mark corresponding multiplier bits values.
2. 00 Or 11: do nothing operation.
3. 01: Marks the ends of a string of 1's and add multiplicand to partial product (running sum).
4. 10: marks the beginning of a string of 1's subtract multiplicand from partial product.

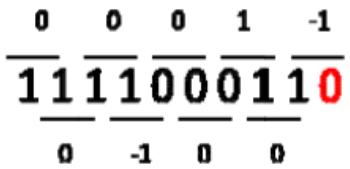

Fig.3: 2 Bit pairing as per booth encoding [6]

The Booth algorithm (Radix-2) has two draw backs which is given below-

- The number of add/subtract and shift operations became variable and hence became inconvenient while designing Parallel multipliers.
- When there are isolated 1's operation the algorithm become inefficient.

These two drawback are overcome by modified booth algorithm (Radix-4).

Modified booth multiplier algorithm:-

The modified booth multiplier was proposed by D.L.Macsorley in 1961. Modified booth algorithm (MBA) is one of the powerful multiplication algorithms [8] for reducing the number of partial products. It is a high speed multiplier used to enhance parallelism which helps to reduce number of partial product row , by using MBA overall the number of partial product are deceased from N to N/2 where N is multiplier[7].

Modified booth algorithm (Radix-4 algorithm):-

The modified-Booth algorithm is extensively used for high-speed multiplier design. By using this technique of Radix- 4 Booth encoding, it is possible to reduce the number of partial products by half. The basic idea of this algorithm is that, instead of adding and shifting for every column of multiplier term and multiplying by 1 or 0, we only consider every second column, and multiply by ±1, ±2, or 0, to obtain the same results. Depending on multiplier bits, radix-4 booth encoder performs the process of encoding the multiplicand. It will compare 3 bits at a time with overlapping technique. First, the grouping starts from the least significant bit (LSB), and the first block only uses 2 bits of multiplier and assumes a zero for the third bit as shown by figure 4.

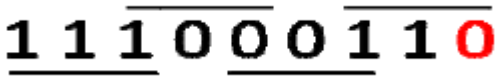

Fig. 4: 3 Bit pairing as per booth encoding [6]

Radix-4 Booth algorithm [9] is given below-

1. Pad the LSB with one zero value.
2. Pad the MSB with 2 zeros if n is even and 1 zero if n is odd.
3. Divide the multiplier into overlapping groups of three-bits.
4. Determine partial product scale factor from modified booth encoding table as shown in table 2 where M is multiplier bits.
5. Compute the Multiplicand Multiples
6. Sum Partial Products

Table 2: Booth Recoding Table for Radix-4[6-9]

M(i+1)	M(i)	M(i-1)	Partial Products
0	0	0	+0*Multiplicand
0	0	1	+1*Multiplicand
0	1	0	+1*Multiplicand
0	1	1	+2* Multiplicand
1	0	0	-2* Multiplicand
1	0	1	-1* Multiplicand
1	1	0	-1* Multiplicand
1	1	1	+0* Multiplicand

Comparison of Radix-2 and Radix-4 algorithm:-

From table 3 after analyzing[6] the two booth multipliers, and compare their characteristics in terms of multiplication speed, number of computations required, number of hardware, it is found that Modified radix 4 booth multiplier is better than Radix-2 booth multiplier.

Table 3: Comparison between Radix-2 and Radix- 4 [6]

Device Utilization Summary	**Radix-2**	**Radix-4**
Number of Slices	397	71
Number of 4 input LUTs	184	100
Number of bonded INPUT	16	16
Number of bonded OUTPUT	16	16
Macro Statistics		
#Latches	24	12
8-bit latches	24	12
#Xors	71	23
1-bit xor2	64	21
8-bit xor2	7	2
Timing Summary		
Minimum period:	5.454ns	4.750ns
Maximum input arrival time before clock:	7.936ns	4.014ns
Maximum output required time after clock:	6.216ns	6.205ns

V. RESULT

Figure 5 shows RTL (Register transfer logic) view of partial product generated in Modified booth stage i.e. first stage of designing a Hybrid multiplier by using a modified booth algorithm.

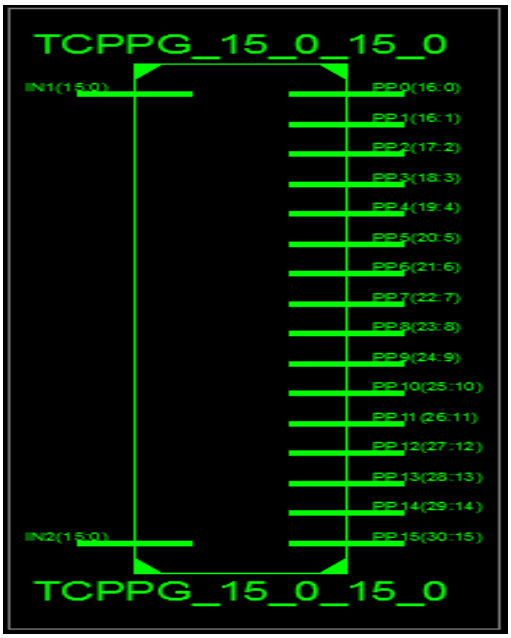

Fig. 5: RTL view of partial product generated in modified booth stage

VI. CONCLUSION

In this paper, we discuss the radix-2 Booth algorithm and radix-4 modified Booth algorithm (MBA) and comparison of both the algorithm. After comparison, it is found that modified Booth algorithm is better than radix-2 Booth algorithm and MBA overcome the drawback of radix-2 booth algorithm. By using this technique of Radix 4 Booth encoding, it is possible to reduce the number of partial products by half. The proposed methodology which consists of modified booth algorithm and Wallace tree structure. Modified Booth algorithm used for reduction of partial product which takes places by using efficient encoding method which save multiplier area and reduce delay at the same time .Wallace tree structure design is used for fast addition of partial products and for final accumulation (i.e. for final addition) either ripple carry adder or carry look ahead adder is used.

VII. REFERENCES

[1] M. N Shanmukha Swamy, Ravindra P Rajput, "High speed modified booth encoder multiplier for signed and unsigned numbers,'' 14th International Conference on Modelling and Simulation, *(IEEE Computer Society*) March 2012.

[2] Nishat Bano, "VLSI Design of Low Power Booth Multiplier," International Journal of Scientific & Engineering Research, *ISSN 2229-5518*, Volume 3, Issue 2, February 2012.

[3] K.Nagarjun, S.Srinivas, "A New Design of Multiplier using Modified Booth Algorithm and Reversible Gate Logic," International Journal of Computer Applications Technology and Research, *ISSN: 2319–8656*, 743 - 747, Volume 2– Issue 6, 2013.

[4] Kulvir Singh, Dilip Kumar, "Modified booth multiplier with carry select adder using 3-stage pipelining technique," International Journal of Computer Applications, Volume 44– No14, April 2012.

[5] A.D. Booth, "A Signed Binary Multiplication Technique," Quarterly Journal of Mechanics and Applied mathematics, Volume-IV, pt-2-1951.

[6] Sukhmeet Kaur, Suman and Manpreet Signh Manna, "Implementation of Modified Booth Algorithm (Radix 4) and its Comparison with Booth Algorithm (Radix-2)," Advance in Electronic and Electric Engineering, *(ISSN 2231-1297)*, Volume 3, Number 6 , pp. 683-690,2013.

[7] A.S.Prabhu, V.Elakya, "Design of modified Low Power Booth Multiplier," *IEEE, 2012*.

[8] Kavita, Jasbir Kaur, "Design and Implementation of an Efficient Modified Booth Multiplier using VHDL," International Conference on Emerging Trend in Engineering and Management, *ISSN: 2231-0347,* Volume.3, 3 July2013.

[9] EPPILI JAYA, K.CHITAMBARA RAO, "Power, area and delay comparison of different multipliers," International Journal of Science, Engineering and Technology Research (IJSETR), *ISSN: 2278 – 7798*, Volume 5, Issue 6, and June 2016.

[10] N.VEDA KUMAR, THEEGALA DHIVYA, "Review of Booth Algorithm for Design of Multiplier," Journal of Emerging Technologies and Innovative Research (JETIR), ***(****ISSN-2349-5162*), Volume 3, Issue 10, October 2016.